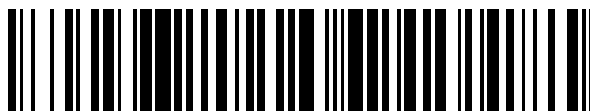


19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 632 623**

51 Int. Cl.:

G06F 17/30

(2006.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

86 Fecha de presentación y número de la solicitud internacional: **10.03.2014** **PCT/US2014/022222**

87 Fecha y número de publicación internacional: **25.09.2014** **WO14150098**

96 Fecha de presentación y número de la solicitud europea: **10.03.2014** **E 14717544 (2)**

97 Fecha y número de publicación de la concesión europea: **19.04.2017** **EP 2973027**

54 Título: **Ejecución de procedimientos almacenados en bases de datos paralelas**

30 Prioridad:

15.03.2013 US 201313831724

45 Fecha de publicación y mención en BOPI de la traducción de la patente:

14.09.2017

73 Titular/es:

MICROSOFT TECHNOLOGY LICENSING, LLC

(100.0%)

One Microsoft Way

Redmond, WA 98052, US

72 Inventor/es:

TELETIA, NIKHIL;

HALVERSON, ALAN DALE y

YEGOROV, ARTEM D.

74 Agente/Representante:

CARPINTERO LÓPEZ, Mario

ES 2 632 623 T3

Aviso: En el plazo de nueve meses a contar desde la fecha de publicación en el Boletín Europeo de Patentes, de la mención de concesión de la patente europea, cualquier persona podrá oponerse ante la Oficina Europea de Patentes a la patente concedida. La oposición deberá formularse por escrito y estar motivada; sólo se considerará como formulada una vez que se haya realizado el pago de la tasa de oposición (art. 99.1 del Convenio sobre Concesión de Patentes Europeas).

DESCRIPCIÓN

Ejecución de procedimientos almacenados en bases de datos paralelas

Antecedentes

1. Antecedentes y técnica relevante

5 Los sistemas informáticos y la tecnología relacionada afectan a muchos aspectos de la sociedad. De hecho, la capacidad del sistema informático para procesar información ha transformado la manera en la que vivimos y trabajamos. Los sistemas informáticos ahora realizan comúnmente una multitud de tareas (por ejemplo, procesamiento de textos, planificación, contabilidad, etc.) que antes de la llegada de los sistemas informáticos se realizaban manualmente. Más recientemente, los sistemas informáticos se han acoplado entre sí y a otros dispositivos electrónicos para formar tanto redes informáticas alámbricas como inalámbricas a través de las cuales los sistemas informáticos y otros dispositivos electrónicos pueden transferir datos electrónicos. Por consiguiente, el rendimiento de muchas tareas informáticas se distribuye a través de un número de diferentes sistemas informáticos y/o un número de diferentes entorno informáticos.

10 En algunos entornos, se emiten consultas contra una recopilación de datos para facilitar la recuperación de información dirigida desde la recopilación de datos. Un usuario (o incluso un sistema informático) formula una consulta usando construcciones de un lenguaje de consultas. Un lenguaje de consultas típicamente incluye un número de construcciones que pueden agruparse en diferentes combinaciones para expresar un intento lógico para recuperar datos. La consulta se emite en un sistema de gestión de datos para procesamiento. El sistema de gestión de datos traduce la consulta en un conjunto correspondiente de operaciones físicas compatibles (en ocasiones y en lo sucesivo denominado como un "plan de consulta") para realizar el intento lógico expresado. El plan de consulta puede a continuación ejecutarse para recuperar datos de la recopilación de datos de acuerdo con el intento lógico expresado. Los datos recuperados pueden devolverse al emisor de la consulta.

15 Por ejemplo, puede usarse SQL para formular una consulta para recuperar datos desde una base de datos relacional. La consulta se emite a un sistema de gestión de base de datos que controla el acceso a la base de datos relacional. El sistema de gestión de base de datos traduce la consulta en un plan de consulta. El plan de consulta se ejecuta a continuación para recuperar datos de la base de datos relacional. Los datos de la base de datos recuperados pueden devolverse al emisor de la consulta.

20 Algunos sistemas de base de datos son sistemas de datos de base de datos independientes (o de nodo único) donde todos los datos y los datos de optimización se almacenan físicamente en la misma máquina. Otros sistemas de bases de datos son sistemas de bases de datos paralelas. En un sistema de bases de datos paralelas, el almacenamiento de la base de datos se distribuye a través de un número de nodos de cálculo. Cada nodo de cálculo almacena una o más porciones de una base de datos localmente. Otros módulos (por ejemplo, en un nodo de control) abstraen la naturaleza distribuida de la base de datos de los usuarios de manera que parece como una única base de datos unificada. Como tal, en un sistema de bases de datos paralelas, los datos relevantes para una consulta así como los datos usados para optimización del plan de consulta pueden distribuirse a través de un número de diferentes nodos.

25 El soporte de la ejecución de procedimientos en lotes y almacenados contra una base de datos paralela tiene al menos un número de dificultades y/o ineficacias. Al menos una dificultad es conservar el comportamiento del sistema único equivalente en un entorno de ejecución de base de datos paralela. El mismo comportamiento puede implementarse en cada nodo de cálculo en una base de datos paralela. Sin embargo, implementar el mismo comportamiento en cada cálculo puede dar como resultado esfuerzo duplicado y realización de operaciones redundantes. La publicación de patente US2003/0014393A1 desvela un procedimiento para ejecución paralela de operaciones de SQL de procedimientos almacenados.

Breve resumen

30 La presente invención se extiende a procedimientos, sistemas y productos de programa informático para ejecutar procedimientos almacenados en bases de datos paralelas. Un coordinador de base de datos paralela está configurado para emitir sentencias de SQL contra una base de datos paralela. La base de datos paralela incluye una pluralidad de particiones de bases de datos distribuidas en una pluralidad de nodos de cálculo. El coordinador de base de datos paralela está también conectado a una instancia de servidor de base de datos independiente.

35 El coordinador de base de datos paralela recibe una solicitud para crear un procedimiento almacenado. La solicitud de creación incluye un nombre y sentencias de procedimiento almacenado. El procedimiento almacenado es para incluir una o más sentencias de SQL. El procedimiento almacenado puede incluir también opcionalmente al menos otra sentencia no de SQL (por ejemplo, flujo de control, asignación variable, evaluación de expresión, etc.). El coordinador de base de datos paralela transforma las sentencias de procedimiento almacenado de modo que la una o más sentencias de SQL se pasan de vuelta al coordinador de base de datos paralela cuando se ejecuta el procedimiento almacenado. La solicitud para crear el procedimiento almacenado se reenvía a continuación al servidor de base de datos independiente. El servidor de base de datos independiente crea y almacena el

procedimiento almacenado transformado.

Posteriormente, el coordinador de base de datos paralela recibe una solicitud de ejecución desde un cliente. La solicitud de ejecución solicita la ejecución del procedimiento almacenado. El coordinador de base de datos paralela reenvía la solicitud de ejecución al servidor de base de datos independiente. El servidor de base de datos independiente ejecuta el procedimiento almacenado transformado. Durante la ejecución en el servidor de base de datos independiente, el procedimiento almacenado transformado envía una solicitud para ejecutar una sentencia de SQL (de entre la una o más sentencias de SQL) al coordinador de base de datos paralela. El coordinador de base de datos paralela recibe la solicitud para ejecutar la sentencia de SQL desde el servidor de base de datos independiente.

El coordinador de base de datos paralela ejecuta la sentencia de SQL contra la base de datos paralela. El coordinador de base de datos paralela recibe resultados a partir de ejecutar la sentencia de SQL contra la base de datos paralela. Dependiendo del tipo de la sentencia de SQL, el coordinador de base de datos paralela devuelve los resultados al servidor de base de datos independiente o al cliente. Cuando sea apropiado, el servidor de base de datos independiente recibe los resultados desde el coordinador de base de datos paralela. El servidor de base de datos independiente usa los resultados durante la ejecución de al menos otra sentencia no de SQL en el servidor de base de datos independiente.

Por consiguiente, la ejecución de sentencias en el procedimiento almacenado transformado se divide entre el coordinador de base de datos paralela y el servidor de base de datos independiente. El coordinador de base de datos paralela ejecuta sentencias de SQL mientras que el servidor de base de datos independiente maneja otras sentencias no de SQL, tales como, sentencias de control de flujo, asignaciones variables, evaluación de expresión, etc.

Este resumen se proporciona para introducir una selección de conceptos en una forma simplificada que se describen adicionalmente a continuación en la descripción detallada. Este resumen no se pretende para identificar características clave o características esenciales de la materia objeto reivindicada, ni se pretende que se use como una ayuda al determinar el alcance de la materia objeto reivindicada.

Características y ventajas adicionales de la invención se expondrán en la descripción que sigue, y en parte serán evidentes a partir de la descripción, o pueden aprenderse mediante la puesta en práctica de la invención. Las características y ventajas de la invención pueden realizarse y obtenerse por medio de los instrumentos y combinaciones particularmente señaladas en las reivindicaciones adjuntas. Estas y otras características de la presente invención serán más completamente evidentes a partir de la siguiente descripción y reivindicaciones adjuntas, o pueden aprenderse mediante la puesta en práctica de la invención como se expone en lo sucesivo.

Breve descripción de los dibujos

Para describir la manera en la que las ventajas y características anteriormente indicadas y otras de la invención pueden obtenerse, se presentará una descripción más particular de la invención brevemente descrita anteriormente por referencia a realizaciones específicas de la misma que se ilustran en los dibujos adjuntos. Entendiendo que estos dibujos representan únicamente realizaciones típicas de la invención y por lo tanto no se han de considerar que son limitantes de su alcance, la invención se describirá y explicará con especificidad adicional y detalle a través del uso de los dibujos adjuntos en los que:

las Figuras 1A y 1B ilustran una arquitectura informática ejemplar que facilita crear y ejecutar procedimientos almacenados en bases de datos paralelas.

La Figura 2 ilustra un diagrama de flujo de un procedimiento ejemplar para crear y ejecutar procedimientos almacenados en bases de datos paralelas.

La Figura 3 ilustra un ejemplo más detallado de dividir la ejecución de procedimiento almacenado entre un servidor de base de datos independiente y un coordinador de base de datos paralela.

Las Figuras 4A-4F ilustran diversos procedimientos en formatos originales y en formatos transformados correspondientes.

Descripción detallada

La presente invención se extiende a procedimientos, sistemas y productos de programa informático para ejecutar procedimientos almacenados en bases de datos paralelas. Un coordinador de base de datos paralela está configurado para emitir sentencias de SQL contra una base de datos paralela. La base de datos paralela incluye una pluralidad de particiones de bases de datos distribuidas en una pluralidad de nodos de cálculo. El coordinador de base de datos paralela está también conectado a una instancia de servidor de base de datos independiente.

El coordinador de base de datos paralela recibe una solicitud para crear un procedimiento almacenado. El procedimiento almacenado incluye una o más sentencias de SQL. El procedimiento almacenado puede incluir también opcionalmente al menos otra sentencia no de SQL (por ejemplo, flujo de control, asignación variable, evaluación de expresión, etc.). El coordinador de base de datos paralela transforma el procedimiento almacenado de modo que la una o más sentencias de SQL se pasan de vuelta al coordinador de base de datos paralela cuando se

ejecuta el procedimiento almacenado. La solicitud para crear el procedimiento almacenado se reenvía a continuación al servidor de base de datos independiente. El servidor de base de datos independiente crea y almacena el procedimiento almacenado transformado.

Posteriormente, el coordinador de base de datos paralela recibe una solicitud de ejecución desde un cliente. La solicitud de ejecución solicita la ejecución del procedimiento almacenado. El coordinador de base de datos paralela reenvía la solicitud de ejecución al servidor de base de datos independiente. El servidor de base de datos independiente ejecuta el procedimiento almacenado transformado. Durante la ejecución en el servidor de base de datos independiente, el procedimiento almacenado transformado envía una solicitud para ejecutar una sentencia de SQL (de entre la una o más sentencias de SQL) al coordinador de base de datos paralela. El coordinador de base de datos paralela recibe la solicitud para ejecutar la sentencia de SQL desde el servidor de base de datos independiente.

El coordinador de base de datos paralela ejecuta la sentencia de SQL contra la base de datos paralela. El coordinador de base de datos paralela recibe resultados a partir de ejecutar la sentencia de SQL contra la base de datos paralela. Dependiendo del tipo de la sentencia de SQL, el coordinador de base de datos paralela devuelve los resultados al servidor de base de datos independiente o al cliente. Cuando sea apropiado, el servidor de base de datos independiente recibe los resultados desde el coordinador de base de datos paralela. El servidor de base de datos independiente usa los resultados para ejecutar la al menos otra sentencia no de SQL en el servidor de base de datos independiente.

Por consiguiente, la ejecución de sentencias en el procedimiento almacenado transformado se divide entre el coordinador de base de datos paralela y el servidor de base de datos independiente. El coordinador de base de datos paralela ejecuta sentencias de SQL mientras el servidor de base de datos independiente maneja otras sentencias no de SQL, tales como, sentencias de control de flujo, asignaciones variables, evaluación de expresión, etc.

Las realizaciones de la presente invención pueden comprender o utilizar un ordenador de fin especial o de fin general que incluye hardware informático, tal como, por ejemplo, uno o más procesadores y memoria de sistema, como se analiza en mayor detalle a continuación. Las realizaciones dentro del alcance de la presente invención también incluyen medios legibles por ordenador físicos y otros para llevar a cabo o almacenar instrucciones y/o estructuras de datos ejecutables por ordenador. Tal medio legible por ordenador puede ser cualquier medio disponible que pueda accederse por un sistema de ordenador de fin general o de fin especial. Medio legible por ordenador que almacena instrucciones ejecutables por ordenador son medios de almacenamiento informático (dispositivos). Medio legible por ordenador que lleva instrucciones ejecutables por ordenador son medios de transmisión. Por lo tanto, a modo de ejemplo, y no como limitación, las realizaciones de la invención pueden comprender al menos dos clases diferentes de manera distinta de medio legible por ordenador: medio de almacenamiento informático (dispositivos) y medio de transmisión.

Medio de almacenamiento informático (dispositivos) incluye RAM, ROM, EEPROM, CD-ROM, unidades de estado sólido ("SSD") (por ejemplo, basadas en RAM), memoria Flash, memoria de cambio de fase ("PCM"), otros tipos de memoria, otro almacenamiento de disco óptico, almacenamiento de disco magnético u otros dispositivos de almacenamiento magnético o cualquier otro medio que pueda usarse para almacenar medios de código de programa deseado en forma de instrucciones o estructuras de datos ejecutables por ordenador y que puedan accederse por un ordenador de fin general o de fin especial.

Una "red" se define como uno o más enlaces de datos que posibilitan el transporte de datos electrónicos entre sistemas informáticos y/o módulos y/u otros dispositivos electrónicos. Cuando la información se transfiere o proporciona a través de una red u otra conexión de comunicaciones (ya sea cableada, inalámbrica o una combinación de cableada o inalámbrica) a un ordenador, el ordenador ve apropiadamente la conexión como un medio de transmisión. Los medios de transmisión pueden incluir una red y/o enlaces de datos que pueden usarse para llevar medios de código de programa deseados en forma de instrucciones o estructuras de datos ejecutables por ordenador y que pueden accederse por un ordenador de fin general o de fin especial. Combinaciones de lo anterior deberían incluirse también dentro del alcance de medio legible por ordenador.

Además, tras alcanzar diversos componentes de sistema informático, medios de código de programa en forma de instrucciones o estructuras de datos ejecutables por ordenador pueden transferirse automáticamente de medio de transmisión a medio de almacenamiento informático (dispositivos) (o viceversa). Por ejemplo, instrucciones o estructuras de datos ejecutables por ordenador recibidas a través de una red o enlace de datos pueden almacenarse en memoria intermedia en RAM dentro de un módulo de interfaz de red (por ejemplo, un "NIC"), y a continuación transferirse eventualmente a RAM de sistema informático y/o a medio de almacenamiento informático menos volátil (dispositivos) en un sistema informático. Por lo tanto, debería entenderse que medio de almacenamiento informático (dispositivos) puede incluirse en componentes de sistema informático que también (o incluso principalmente) utilizan el medio de transmisión.

Instrucciones ejecutables por ordenador comprenden, por ejemplo, instrucciones y datos que, cuando se ejecutan en un procesador, provocan que un ordenador de fin general, ordenador de fin especial o dispositivo de procesamiento

de fin especial realice una cierta función o grupo de funciones. Las instrucciones ejecutables por ordenador pueden ser, por ejemplo, binarios, instrucciones de formato intermedio tal como lenguaje ensamblador o incluso código fuente. Aunque se ha descrito la materia objeto en lenguaje específico a características estructurales y/o actos metodológicos, se ha de entender que la materia objeto definida en las reivindicaciones adjuntas no está necesariamente limitada a las características descritas o actos anteriormente descritos. En su lugar, las características y actos descritos se desvelan como formas de ejemplo para implementar las reivindicaciones.

Los expertos en la materia apreciarán que la invención puede ponerse en práctica en entornos informáticos de red con muchos tipos de configuraciones de sistema informático, incluyendo, ordenadores personales, ordenadores de sobremesa, ordenadores portátiles, procesadores de mensajes, dispositivos portátiles, sistemas multiprocesador, electrónica de consumo basada en microprocesador o programable, PC de red, miniordenadores, ordenadores centrales, teléfonos móviles, PDA, tabletas, buscapersonas, encaminadores, conmutadores y similares. La invención también puede ponerse en práctica en entornos de sistemas distribuidos donde sistemas informáticos locales y remotos que están vinculados (ya sean enlaces de datos cableados, enlaces de datos inalámbricos o mediante una combinación de enlaces de datos cableados e inalámbricos) a través de una red, ambos realizan tareas. En un entorno de sistema distribuido, los módulos de programa pueden localizarse tanto en dispositivos de almacenamiento de memoria locales como remotos.

Las realizaciones de la invención pueden implementarse también en entornos informáticos en la nube. En esta descripción y en las siguientes reivindicaciones, "informática en la nube" se define como un modelo para posibilitar acceso de red bajo demanda a una agrupación compartida de recursos informáticos configurables. Por ejemplo, informática en la nube puede emplearse en el mercado para ofrecer acceso bajo demanda generalizado y cómodo a la agrupación compartida de recursos informáticos configurables. La agrupación compartida de recursos informáticos configurables puede aprovisionarse rápidamente mediante virtualización y liberarse con bajo esfuerzo de gestión o interacción del proveedor de servicio, y a continuación cambiarse de escala en consecuencia.

Un modelo informático en la nube puede estar compuesto de características tales como, por ejemplo, auto-servicio bajo demanda, acceso de red amplio, agrupación de recursos, elasticidad rápida, servicio medido y así sucesivamente. Un modelo informático en la nube puede exponer también diversos modelos de servicio, tales como, por ejemplo, Software como un Servicio ("SaaS"), Plataforma como un servicio ("PaaS") e Infraestructura como un Servicio ("IaaS"). Un modelo informático en la nube puede desplegarse también usando diferentes modelos de despliegue tales como nube privada, nube comunitaria, nube pública, nube híbrida y así sucesivamente. En esta descripción y en las reivindicaciones, un "entorno informático en la nube" es un entorno en el que se emplea informática en la nube.

En general, las realizaciones de la invención transforman un procedimiento almacenado de modo que la ejecución del procedimiento almacenado se divide entre un servidor de base de datos independiente (por ejemplo, una base de datos de intérprete de comandos) y un coordinador de base de datos paralela (por ejemplo, un motor de almacén de datos paralelo (PDW)). La ejecución del procedimiento almacenado se inicia en el servidor de base de datos independiente. En tiempo de ejecución, las sentencias de flujo de control, asignación variable, evaluación de expresión, etc., se manejan mediante (o están centralizadas en) el servidor de base de datos independiente. Las sentencias de SQL (por ejemplo, SELECT, CREATE, DELETE, UPDATE, INSERT, etc.) se pasan desde el servidor de base de datos independiente al coordinador de base de datos paralela para la ejecución. Los resultados de las sentencias de SQL ejecutadas pueden devolverse al servidor de base de datos independiente o a un cliente. El coordinador de base de datos paralela puede añadirse como un servidor vinculado al servidor de base de datos independiente. Las sentencias, tales como, por ejemplo, OPENQUERY, EXECUTE AT, pueden usarse para pasar SQL del servidor de base de datos independiente al coordinador de base de datos paralela. En algunas realizaciones, se usa un testigo de sesión para compartir el estado de sesión entre diferentes partes.

Las Figuras 1A y 1B ilustran una arquitectura 100 informática de ejemplo que facilita crear y ejecutar procedimientos almacenados en bases de datos paralelas. Haciendo referencia a la Figura 1A, la arquitectura 100 informática incluye el cliente 101, el coordinador 102 de base de datos paralela, el servidor 103 de base de datos independiente, la base de datos 141 y el almacenamiento 107. Cada uno del cliente 101, coordinador 102 de base de datos paralela, servidor 103 de base de datos independiente, base de datos 141 y almacenamiento 107 pueden estar conectados entre sí a través de (o ser parte de) una red, tal como, por ejemplo, una red de área local ("LAN"), una red de área extensa ("WAN"), e incluso internet. Por consiguiente, el cliente 101, coordinador 102 de base de datos paralela, servidor 103 de base de datos independiente, base de datos 141 y almacenamiento 107 así como cualquier otro sistema informático conectado y sus componentes, pueden crear datos relacionados con el mensaje e intercambiar datos relacionados con el mensaje (por ejemplo, datagramas del Protocolo de Internet ("IP") y otros protocolos de capa superior que utilizan datagramas de IP, tales como, el Protocolo de Control de Transmisión ("TCP"), Protocolo de Transferencia de Hipertexto ("HTTP"), Protocolo de Transferencia de Correo Simple ("SMTP"), etc., o usando otros protocolos no de datagramas) a través de la red.

Haciendo referencia ahora a la Figura 1A, el cliente 101 puede ser un cliente de base de datos. El cliente 101 puede proporcionar una interfaz de usuario para interactuar con datos almacenados en la base de datos 131. El cliente 101 puede enviar solicitudes para crear procedimientos almacenados al coordinador 102 de base de datos paralela. Las solicitudes de creación de procedimiento almacenado pueden incluir un nombre para usar para el procedimiento

almacenado y sentencias de procedimiento almacenado que se han de ejecutar.

El coordinador 102 de base de datos paralela (por ejemplo, un motor de Almacén de Datos Paralelo (PDW)) puede recibir solicitudes desde el cliente 101. Cuando se recibe una solicitud de creación de procedimiento almacenado, el módulo 104 de transformación de procedimiento puede transformar sentencias de procedimiento almacenado de la solicitud. La transformación puede provocar que cualquier sentencia de SQL en las sentencias de procedimiento almacenado se devuelva al coordinador 102 de base de datos paralela para procesar cuando se ejecuta el procedimiento almacenado. El coordinador 102 de base de datos paralela puede a continuación enviar la solicitud de creación de procedimiento almacenado con el nombre y las sentencias de procedimiento almacenado transformado al servidor 103 de base de datos independiente.

El servidor 130 de base de datos independiente (por ejemplo, una base de datos de intérprete de comandos) puede recibir la solicitud de creación de procedimiento almacenado. El módulo 103 de creación de procedimiento almacenado puede crear un procedimiento ejecutable desde las sentencias de procedimiento almacenado transformado. El servidor 130 de base de datos independiente puede almacenar el nombre junto con el procedimiento ejecutable en el almacenamiento 107.

La base de datos 131 incluye las porciones 131A, 131B y 131C de base de datos y los nodos 121, 122 y 123 de cálculo. Los nodos 121, 122 y 123 de cálculo pueden configurarse en una arquitectura nada compartida. Como tal, cada uno de los nodos 121, 122 y 123 de cálculo mantienen memoria separada (por ejemplo, RAM) y recursos de almacenamiento separados (por ejemplo, unidades de disco) de otros nodos. La base de datos 131 está distribuida a través del nodo 121, 122 y 123 informático, que puede mantener las porciones 131A, 131B y 131C de base de datos respectivamente. El coordinador 102 de base de datos paralela puede incluir hardware de abstracción y/o componentes de software que abstraen la naturaleza distribuida de la base de datos 131. El coordinador 109 de base de datos paralela puede usar el hardware de abstracción y/o componentes de software para presentar la base de datos 131 como una base de datos independiente al cliente 101. Por lo tanto, el cliente 101 puede interactuar con la base de datos 131 usando comandos compatibles con una base de datos independiente (incluyendo enviar consultas).

La Figura 2 ilustra un diagrama de flujo de un procedimiento 200 de ejemplo para crear y ejecutar procedimientos almacenados en bases de datos paralelas. El procedimiento 200 se describirá con respecto a los componentes y datos de la arquitectura 100 informática.

El procedimiento 200 incluye enviar una solicitud de creación para crear un procedimiento almacenado a un coordinador de base de datos paralela, incluyendo la solicitud de creación un nombre y sentencias de procedimiento almacenado, el procedimiento almacenado para incluir una o más sentencias (201) de SQL. El procedimiento almacenado puede incluir también opcionalmente al menos una sentencia no de SQL. Por ejemplo, el cliente 101 puede enviar la solicitud 111 creada al coordinador 102 de base de datos paralela. La solicitud 111 de creación contiene el nombre 112 y las sentencias 113 de procedimiento almacenado. El nombre 112 indica el nombre a proporcionarse al procedimiento almacenado. Las sentencias 113 de procedimiento almacenado se han de usar para crear un procedimiento ejecutable. Las sentencias 113 de procedimiento almacenado pueden incluir sentencias de SQL y potencialmente también sentencias no de SQL (por ejemplo, sentencias de flujo de control, asignaciones variables, evaluaciones de expresión, etc.). El procedimiento 200 incluye recibir la solicitud de creación desde el cliente (202). Por ejemplo, el coordinador 102 de base de datos paralela puede recibir la solicitud 111 de creación desde el cliente 101.

El procedimiento 200 incluye transformar las sentencias de procedimiento almacenado de modo que la una o más sentencias de SQL se pasan de vuelta al coordinador de base de datos paralela cuando se ejecuta el procedimiento almacenado (203). Por ejemplo, el módulo 104 de transformación de procedimiento puede transformar las sentencias 113 de procedimiento almacenado en sentencias 113T de procedimiento almacenado transformado. Cualquier sentencia de SQL en las sentencias 113 de procedimiento almacenado puede transformarse en sentencias 113T de procedimiento almacenado transformado de modo que las sentencias de SQL se pasan de vuelta al coordinador de base de datos paralela 103 durante la ejecución.

Las Figuras 4A-4F representan diversas transformaciones que facilitan transferir sentencias de SQL de vuelta a un coordinador de base de datos paralela. En algunas realizaciones, las sentencias SELECT se transforman usando cualquiera de una sentencia EXECUTE o una sentencia OPENQUERY. Un coordinador de base de datos paralela puede añadirse en las sentencias de procedimiento almacenado como una base de datos vinculada (por ejemplo, "[LINKED DB]") de modo que las sentencias EXECUTE u OPENQUERY se pasan de vuelta al coordinador de base de datos paralela durante la ejecución.

La Figura 4A representa dos posibles transformaciones para la consulta SELECT sin ningún parámetro. La Figura 4B representa una posible transformación para la consulta SELECT con parámetros. La Figura 4C representa dos posibles transformaciones para la consulta SELECT con asignaciones variables pero no parámetros. La Figura 4D representa una posible transformación para la subconsulta escalar sin parámetros. La Figura 4E representa una posible transformación para una consulta escalar en una sentencia de flujo de control. La Figura 4F representa una posible transformación para una consulta SELECT con una sentencia INTO sin ningún parámetro.

Cualquier número de otras transformaciones es también posible. En algunas realizaciones, las transformaciones pueden facilitarse extendiendo la gramática aceptada por un analizador de coordinador de base de datos paralela. En otras realizaciones, OPENQUERY puede extenderse para aceptar parámetros.

Volviendo a la Figura 2, el procedimiento 200 incluye reenviar una solicitud de creación adicional a un servidor de base de datos independiente, incluyendo la solicitud de creación adicional el nombre y las sentencias (204) de procedimiento almacenado transformado. Por ejemplo, el coordinador 102 de base de datos paralela puede reenviar la solicitud 114 de creación al servidor 103 de base de datos independiente. Como se representa, la solicitud 114 de creación incluye el nombre 112 y las sentencias 113T de procedimiento almacenado transformado. El procedimiento 200 incluye recibir la solicitud de creación adicional desde el coordinador (205) de base de datos paralela. Por ejemplo, el servidor 103 de base de datos independiente puede recibir la solicitud 114 creada.

El procedimiento 200 incluye crear y almacenar un procedimiento almacenado desde las sentencias (206) de procedimiento almacenado transformado. Por ejemplo, el módulo 106 de creación de procedimiento almacenado puede crear el procedimiento 116 ejecutable desde sentencias 113T de procedimiento almacenado transformado. El procedimiento 116 ejecutable incluye código ejecutable que representa las sentencias 117 de SQL y potencialmente que también representa sentencias 118 no de SQL. El servidor 103 de base de datos independiente puede almacenar el procedimiento 116 ejecutable junto con el nombre 112 en el almacenamiento 107.

Volviendo a la Figura 1B, el cliente 101 puede enviar también solicitudes para ejecutar procedimientos almacenados al coordinador 102 de base de datos paralela. Las solicitudes de ejecución de procedimiento almacenado pueden incluir el nombre de un procedimiento almacenado que se ha de ejecutar. El coordinador 102 de base de datos paralela puede recibir la solicitud de ejecución del procedimiento almacenado desde el cliente 101. En respuesta, el coordinador 102 de base de datos paralela puede reenviar la solicitud de ejecución del procedimiento almacenado, que incluye el nombre de un procedimiento almacenado que se ha de ejecutar, al servidor 103 de base de datos independiente.

Cuando sea apropiado, el módulo 152 de correlación de sesión puede generar un testigo de sesión. El coordinador 102 de base de datos paralela puede anexar el testigo de sesión a la solicitud de ejecución de procedimiento almacenado renviado. El testigo de sesión puede usarse para compartir el estado de sesión que el cliente puede tener, incluyendo transacciones, contexto de base de datos, etc., así como autorizar la solicitud sin tener que almacenar y pasar credenciales de usuario al coordinador de base de datos paralela cuando el procedimiento almacenado se ejecuta posteriormente.

El servidor 103 de base de datos independiente puede recibir la solicitud de ejecución de procedimiento almacenado desde el coordinador 102 de base de datos paralela. El módulo 153 de ejecución de procedimiento puede hacer referencia al almacenamiento 107 para identificar un procedimiento almacenado que tiene el nombre incluido. El procedimiento almacenado puede ser un procedimiento almacenado creado desde sentencias de procedimiento almacenado transformado de manera que cualquier sentencia de SQL se pasa al coordinador 102 de base de datos paralela durante la ejecución. El módulo 153 de ejecución de procedimiento puede iniciar la ejecución del procedimiento almacenado en el servidor 103 de base de datos independiente.

Durante la ejecución, las sentencias no de SQL (por ejemplo, flujo de control, asignaciones variables, evaluación de expresión, etc.) se ejecutan en el servidor 103 de base de datos independiente. Cuando las sentencias de SQL (por ejemplo, adjuntas en EXECUTE AT u OPENQUERY) se encuentran, las sentencias de SQL se reenvían al coordinador 102 de base de datos paralela (basándose en el coordinador 102 de base de datos paralela que es una base de datos vinculada). Cuando sea apropiado, un testigo de sesión (recibido previamente desde el coordinador 102 de base de datos paralela) puede anexarse a cualquier sentencia de SQL pasada al coordinador 102 de base de datos paralela.

Como tal, durante la ejecución de un procedimiento almacenado, el coordinador 102 de base de datos paralela puede recibir sentencias de SQL, posiblemente junto con un testigo de sesión, desde el servidor 103 de base de datos independiente. Cuando se recibe un testigo de sesión, el módulo 152 de correlación de sesión puede usar el testigo de sesión para alinear la ejecución de la sentencia de SQL con el estado para el cliente 101. Es decir, la sentencia de SQL puede ejecutarse bajo la misma sesión que la solicitud de ejecución de procedimiento almacenado originalmente recibida desde el cliente 101. El procesador 151 de sentencia de SQL puede ejecutar la sentencia de SQL para enviar una consulta a la base de datos 131. La base de datos 131 puede devolver resultados de consulta de vuelta al coordinador 102 de base de datos paralela.

Los resultados de consulta pueden devolverse al servidor 103 de base de datos independiente o pueden devolverse al cliente 101. Ya se devuelvan los resultados al servidor 103 de base de datos independiente o al cliente 101, puede basarse en el tipo de la sentencia de SQL. Por ejemplo, puede devolverse un resultado de subconsulta escalar al servidor 103 de base de datos independiente. Por otra parte, puede devolverse un resultado de sentencia SELECT normal al cliente 101.

El módulo 153 de ejecución de procedimiento puede usar resultados devueltos durante la ejecución de otras sentencias no de SQL, tal como, por ejemplo, para realizar decisiones de flujo de control, asignar un valor a una

variable, o evaluar una expresión.

Durante la ejecución de un procedimiento almacenado, puede establecerse cualquier número de sentencias de SQL al coordinador 102 de base de datos paralela y devolverse los resultados al servidor 103 de base de datos independiente o al cliente 101.

- 5 Volviendo de nuevo a la Figura 2, el procedimiento 200 incluye enviar una sentencia de ejecución al coordinador de base de datos paralela, solicitando la solicitud de ejecución la ejecución del procedimiento (207) almacenado. Por ejemplo, el cliente 101 puede enviar la solicitud 141 de ejecución al coordinador 102 de base de datos paralela. La solicitud 141 de ejecución puede solicitar la ejecución de un procedimiento almacenado con el nombre 112. El procedimiento 200 incluye recibir la solicitud de ejecución desde el cliente (208). Por ejemplo, el coordinador 102 de base de datos paralela puede recibir la solicitud 141 de ejecución desde el cliente 101.

- 10 El procedimiento 200 incluye reenviar la solicitud de ejecución al servidor (209) de base de datos independiente. Por ejemplo, el coordinador 102 de base de datos paralela puede analizar la solicitud 141 de ejecución y reenviar la correspondiente solicitud 142 de ejecución al servidor 103 de base de datos independiente. Cuando sea apropiado, el módulo 152 de correlación de sesión puede generar un testigo 143 de sesión para la sesión entre el cliente 101 y el coordinador 102 de base de datos paralela. El módulo 152 de correlación de sesión puede anexar el testigo 143 de sesión a la solicitud 142 de ejecución. El procedimiento 200 incluye recibir la solicitud de ejecución desde el coordinador (210) de base de datos paralela. Por ejemplo, el servidor 130 de base de datos independiente puede recibir la solicitud 142 de ejecución desde el coordinador 102 de base de datos paralela.

- 15 El procedimiento 200 incluye ejecutar el procedimiento (211) almacenado. Por ejemplo, en respuesta a recibir la solicitud 142 de ejecución, el módulo 153 de ejecución de procedimiento puede determinar que el nombre 112 corresponde al procedimiento 116 ejecutable. Como tal, el módulo 153 de ejecución de procedimiento puede iniciar la ejecución del procedimiento 116 ejecutable. La ejecución de las sentencias 117 de SQL y las sentencias 118 no de SQL puede dividirse entre el coordinador 102 de base de datos paralela y el servidor 103 de base de datos independiente respectivamente.

- 20 El procedimiento 200 incluye enviar una solicitud para ejecutar una sentencia de SQL al coordinador (212) de base de datos paralela. Durante la ejecución del procedimiento 116 ejecutable, pueden encontrarse las sentencias 117 de SQL y las sentencias 118 no de SQL. Las sentencias de SQL encontradas (por ejemplo, envueltas dentro de EXECUTE AT u OPENQUERY) pueden enviarse al coordinador 102 de base de datos paralela para ejecución. Por ejemplo, el módulo 153 de ejecución de procedimiento puede enviar la sentencia 117A de SQL (una sentencia de SQL incluida en las sentencias 117 de SQL) al coordinador 102 de base de datos paralela. Cuando sea apropiado, el testigo 143 de sesión puede enviarse junto con la sentencia 117A de SQL al coordinador 102 de base de datos paralela.

- 25 El procedimiento 200 incluye recibir la solicitud para ejecutar la sentencia de SQL desde el servidor (213) de base de datos independiente. Por ejemplo, el coordinador 102 de base de datos paralela puede recibir la sentencia 117A de SQL desde el servidor 103 de base de datos independiente. Cuando sea apropiado, el coordinador 102 de base de datos paralela puede recibir el testigo 143 de sesión junto con la sentencia 117A de SQL. El módulo 152 de correlación de sesión puede usar el testigo 143 de sesión para alinear la sentencia 117A de SQL a la sesión entre el cliente 101 y el coordinador 102 de base de datos paralela.

- 30 El procedimiento 200 incluye ejecutar la sentencia de SQL contra una base (214) de datos paralela. Por ejemplo, el procesador 151 de sentencia de SQL puede ejecutar la sentencia 117A de SQL contra la base de datos 131. Ejecutar la sentencia 117A de SQL puede incluir que el coordinador 102 de base de datos paralela emita la consulta 144 a la base de datos 131. El procedimiento 200 incluye recibir resultados a partir de ejecutar la sentencia de SQL contra la base de datos (215) paralela. Por ejemplo, coordinador 102 de base de datos paralela puede recibir resultados 146 (para la consulta 144) a partir de ejecutar la sentencia 117A de SQL contra la base de datos 131.

- 35 El procedimiento 200 incluye devolver los resultados a un destino apropiado basándose en el tipo de la sentencia (216) de SQL. Por ejemplo, el coordinador 102 de base de datos paralela puede devolver los resultados 146 al servidor 103 de base de datos independiente. Como alternativa, los resultados 146 pueden devolverse al cliente 101. Ya se devuelvan los resultados a la base de datos independiente 103 o al cliente 101 puede ser dependiendo del tipo de sentencia 117A de SQL. Por ejemplo, si la sentencia 117A de SQL devuelve un resultado de subconsulta escalar, los resultados 146 pueden devolverse al servidor 103 de base de datos independiente. Por otra parte, si la sentencia 117A de SQL es una sentencia SELECT normal, los resultados 146 pueden devolverse al cliente 101.

- 40 El procedimiento 200 incluye recibir los resultados desde el coordinador de base de datos paralela (en el servidor de base de datos independiente) (217). Por ejemplo, el servidor 103 de base de datos independiente puede recibir resultados 146 desde el coordinador 102 de base de datos paralela. El procedimiento 200 incluye usar los resultados durante la ejecución de al menos otra sentencia (218) no de SQL. Por ejemplo, el módulo 153 de ejecución de procedimiento puede usar los resultados 146 durante la ejecución de al menos una de las sentencias 118 no de SQL. Como alternativa, el procedimiento 200 incluye recibir los resultados desde el coordinador de base de datos paralela (en el cliente) (219). Por ejemplo, el cliente 101 puede recibir los resultados 146.

Según sea apropiado, de 212 a 218 pueden repetirse para cualesquiera otras sentencias de SQL incluidas en las sentencias 117 de SQL.

La Figura 3 ilustra un ejemplo más detallado de dividir la ejecución del procedimiento almacenado entre el servidor 103 de base de datos independiente y un coordinador 102 de base de datos paralela. Como se representa, el módulo 153 de ejecución de procedimiento es el procedimiento 116 de ejecución. Durante la ejecución, la sentencia 301 de SQL se envía al coordinador 102 de base de datos paralela. El procesador 151 de sentencia de SQL procesa la sentencia 301 de SQL para enviar la consulta 302 a la base de datos 131. La consulta 302 puede solicitar el número de entradas en una tabla de cliente. El procesador de sentencia de SQL puede recibir resultados 303 de vuelta en respuesta a una consulta 302 emitida. Los resultados 303 pueden incluir el número de entradas en la tabla de cliente.

El procesador 151 de sentencia de SQL puede devolver resultados 303 al módulo 153 de ejecución de procedimiento. El módulo 153 de ejecución de procedimiento puede recibir los resultados 303 desde el procesador 151 de sentencia de SQL. El módulo 153 de ejecución de procedimiento puede usar los resultados 303 al evaluar la sentencia 304 condicional en el servidor 103 de base de datos independiente. Como tal, durante la ejecución del procedimiento 116, la sentencia 301 de SQL se ejecuta en el coordinador 102 de base de datos paralela y la sentencia 304 condicional se ejecuta en la base de datos 103 independiente.

Para soportar el comportamiento específico del procedimiento almacenado adicional, las realizaciones de la invención pueden usar un puerto de devolución de llamada especializado alojado por el coordinador 102 de base de datos paralela. El puerto de devolución de llamada se aloja exponiendo un punto de extremo de coordinador de base de datos paralela internamente a un Servidor de SQL de nodo de control. El servidor de SQL de nodo de control tiene un servidor vinculado registrado que se conecta a ese punto de extremo mediante su proveedor OLEDB. La exposición del puerto puede limitarse al límite del nodo de control de modo que no se usa autenticación de cliente para el servidor vinculado para comunicar el puerto. En su lugar, puede generarse un testigo de sesión en el momento de la creación de sesión de usuario y puede anexarse a cualquier consulta enviada desde el Servidor de SQL al procedimiento de motor mediante el servidor vinculado.

Puesto que se expone un límite de puerto diferente, la responsabilidad del anfitrión detrás del puerto puede extenderse para poder quitar el testigo de sesión de una consulta entrante, validar la consulta y anexar la ejecución de la consulta a la sesión asociada. Como tal, cualquier estado de sesión que pueda tener un usuario puede compartirse, incluyendo las transacciones, contexto de base de datos, etc. El testigo de sesión puede usarse también para autorizar la solicitud sin tener que almacenar y pasar credenciales de usuario al servidor vinculado todo el tiempo. El testigo de sesión puede estar dispuesto en el final de la sesión del usuario. La lógica para analizar testigos de sesión puede desactivarse para las solicitudes externas y posibilitarse para el puerto interno.

La presente invención puede realizarse en otras formas específicas sin alejarse de sus características esenciales. Las realizaciones descritas se han de considerar en todos los aspectos únicamente como ilustrativas y no restrictivas. El alcance de la invención se indica, por lo tanto, mediante las reivindicaciones adjuntas en su lugar mediante la siguiente descripción. Todos los cambios que entran dentro del significado y alcance de equivalencia de las reivindicaciones se han de abarcar dentro de su alcance.

REIVINDICACIONES

1. Un procedimiento de ejecución de un procedimiento almacenado en un sistema informático, incluyendo el sistema informático un coordinador de base de datos paralela, el coordinador de base de datos paralela configurado para emitir sentencias de SQL contra una base de datos paralela, incluyendo la base de datos paralela una pluralidad de particiones de bases de datos distribuidas en una pluralidad de nodos de cálculo, el coordinador de base de datos paralela conectado a una instancia de servidor de base de datos independiente, comprendiendo el procedimiento:
 - 5 recibir una solicitud de ejecución, solicitando la solicitud de ejecución la ejecución de un procedimiento almacenado creado previamente;
 - reenviar la solicitud de ejecución al servidor de base de datos independiente;
 - 10 recibir una solicitud desde el servidor de base de datos independiente para ejecutar una sentencia de SQL contra la base de datos paralela, la solicitud emitida desde un procedimiento almacenado transformado correspondiente que se ejecuta en el servidor de base de datos independiente en respuesta a recibir la solicitud de ejecución, incluyendo el procedimiento almacenado transformado una o más sentencias de SQL y al menos otra sentencia no de SQL, manejando el servidor de base de datos independiente la ejecución de la al menos otra sentencia no de SQL, habiéndose transformado previamente el procedimiento almacenado transformado para pasar la
 - 15 sentencia de SQL de vuelta al coordinador de base de datos paralela envolviendo la sentencia de SQL en otra sentencia y haciendo referencia al coordinador de base de datos paralela como un servidor vinculado;
 - ejecutar la sentencia de SQL contra la base de datos paralela;
 - recibir resultados a partir de ejecutar la sentencia de SQL contra la base de datos paralela; y
 - 20 devolver los resultados a un destino apropiado basándose en el tipo de la sentencia de SQL.
2. El procedimiento de la reivindicación 1, que comprende adicionalmente antes de recibir la solicitud de ejecución:
 - recibir una solicitud de creación para crear el procedimiento almacenado, incluyendo la solicitud de creación un nombre y sentencias de procedimiento almacenado, incluyendo las sentencias de procedimiento almacenado una o más sentencias de SQL y al menos otra sentencia no de SQL;
 - 25 transformar las sentencias de procedimiento almacenado de modo que la una o más sentencias de SQL se pasan de vuelta al coordinador de base de datos paralela cuando se ejecuta el procedimiento almacenado; y
 - reenviar una solicitud de creación adicional a un servidor de base de datos independiente, incluyendo la solicitud de creación adicional el nombre y las sentencias de procedimiento almacenado transformado.
3. El procedimiento de la reivindicación 1, que comprende adicionalmente crear un testigo de sesión para la solicitud de ejecución; y
- 30 en el que reenviar la solicitud de ejecución al servidor de base de datos independiente comprende reenviar la solicitud de ejecución junto con el testigo de sesión al servidor de base de datos independiente.
4. El procedimiento de la reivindicación 1, en el que recibir una solicitud de ejecución comprende recibir una solicitud desde un cliente de base de datos; y
- 35 en el que devolver los resultados a un destino apropiado basándose en el tipo de la sentencia de SQL comprende devolver los resultados al cliente de base de datos.
5. El procedimiento de la reivindicación 1, en el que devolver los resultados a un destino apropiado basándose en el tipo de la sentencia de SQL comprende devolver los resultados al servidor de base de datos independiente.
6. Un producto de programa informático para uso en un sistema informático, incluyendo el sistema informático un coordinador de base de datos paralela, el coordinador de base de datos paralela configurado para emitir sentencias de SQL contra una base de datos paralela, incluyendo la base de datos paralela una pluralidad de particiones de bases de datos distribuidas en una pluralidad de nodos de cálculo, el coordinador de base de datos paralela conectado a una instancia de servidor de base de datos independiente, el producto de programa informático para implementar un procedimiento para ejecutar un procedimiento almacenado, comprendiendo el producto de programa
- 40 informático uno o más dispositivos de almacenamiento informático que tienen almacenados en los mismos instrucciones ejecutables por ordenador que, cuando se ejecutan en un procesador, provocan que el sistema informático realice el procedimiento, que incluye lo siguiente:
 - recibir una solicitud de ejecución, solicitando la solicitud de ejecución la ejecución de un procedimiento almacenado creado previamente;
 - 50 reenviar la solicitud de ejecución al servidor de base de datos independiente;
 - recibir una solicitud desde el servidor de base de datos independiente para ejecutar una sentencia de SQL contra la base de datos paralela, la solicitud emitida desde un procedimiento almacenado transformado correspondiente que se ejecuta en el servidor de base de datos independiente en respuesta a recibir la solicitud de ejecución, incluyendo el procedimiento almacenado transformado una o más sentencias de SQL y al menos otra sentencia
 - 55 no de SQL, manejando el servidor de base de datos independiente la ejecución de la al menos otra sentencia no de SQL, habiéndose transformado previamente el procedimiento almacenado transformado para pasar la sentencia de SQL de vuelta al coordinador de base de datos paralela envolviendo la sentencia de SQL en otra sentencia y haciendo referencia al coordinador de base de datos paralela como un servidor vinculado;

ejecutar la sentencia de SQL contra la base de datos paralela;
 recibir resultados a partir de ejecutar la sentencia de SQL contra la base de datos paralela; y
 devolver los resultados a un destino apropiado basándose en el tipo de la sentencia de SQL.

- 5 7. El producto de programa informático de la reivindicación 6, que comprende adicionalmente instrucciones ejecutables por ordenador que, cuando se ejecutan, provocan que el sistema informático realice antes de recibir la solicitud de ejecución:

recibir una solicitud de creación para crear el procedimiento almacenado, incluyendo la solicitud de creación un nombre y sentencias de procedimiento almacenado, incluyendo las sentencias de procedimiento almacenado una o más sentencias de SQL y al menos otra sentencia no de SQL;

- 10 transformar las sentencias de procedimiento almacenado de modo que la una o más sentencias de SQL se pasan de vuelta al coordinador de base de datos paralela cuando se ejecuta el procedimiento almacenado, que incluye:

envolver la una o más sentencias de SQL dentro de una de: EXECUTE AT y OPENQUERY; y
 hacer referencia al coordinador de base de datos paralela como un servidor vinculado; y

- 15 reenviar una solicitud de creación adicional al servidor de base de datos independiente, incluyendo la solicitud de creación adicional el nombre y las sentencias de procedimiento almacenado transformado.

8. El producto de programa informático de la reivindicación 6, en el que las instrucciones ejecutables por ordenador que, cuando se ejecutan, provocan que el sistema informático reciba una solicitud desde el servidor de base de datos independiente comprende, en el que las instrucciones ejecutables por ordenador que, cuando se ejecutan,
 20 provocan que el sistema informático reciba una solicitud desde el servidor de base de datos independiente para ejecutar una sentencia de SQL seleccionada de entre: SELECT, CREATE, DELETE, UPDATE e INSERT.

9. Un sistema, incluyendo el sistema el coordinador de base de datos paralela y un servidor de base de datos independiente, comprendiendo el coordinador de base de datos paralela:

- 25 uno o más procesadores;
 memoria de sistema; y
 uno o más dispositivos de almacenamiento informático que tienen almacenados en los mismos instrucciones ejecutables por ordenador que representan un procesador de sentencia de SQL y un módulo de correlación de sesión, el procedimiento de sentencia de SQL configurado para:

- 30 recibir una solicitud desde el servidor de base de datos independiente para ejecutar una sentencia de SQL contra una base de datos paralela, la solicitud emitida desde un procedimiento almacenado transformado correspondiente que se ejecuta en el servidor de base de datos independiente en respuesta a recibir una solicitud de ejecución que solicita la ejecución de un procedimiento almacenado creado previamente, incluyendo el procedimiento almacenado transformado una o más sentencias de SQL y al menos otra
 35 sentencia no de SQL, manejando el servidor de base de datos independiente la ejecución de la al menos otra sentencia no de SQL, habiéndose transformado previamente el procedimiento almacenado transformado para pasar la sentencia de SQL de vuelta al coordinador de base de datos paralela envolviendo la sentencia de SQL en otra sentencia y haciendo referencia al coordinador de base de datos paralela como un servidor vinculado;

- 40 ejecutar la sentencia de SQL contra la base de datos paralela;
 recibir resultados a partir de ejecutar la sentencia de SQL contra la base de datos paralela; y
 devolver los resultados a unos destinos apropiados basándose en el tipo de la sentencia de SQL; y
 en el que el módulo de correlación de sesión está configurado para:

- 45 recibir una solicitud de ejecución;
 crear testigos de sesión para la solicitud de ejecución recibida;
 recibir de vuelta un testigo de sesión creado junto con una solicitud recibida desde el servidor de base de datos independiente; y
 usar el testigo de sesión creado para alinear la solicitud recibida con las sesiones de cliente.

10. El sistema de la reivindicación 9, en el que la base de datos independiente comprende:

- 50 uno o más procesadores;
 memoria de sistema; y
 uno o más dispositivos de almacenamiento informático que tienen almacenados en los mismos instrucciones ejecutables por ordenador que representan un módulo de ejecución de procedimiento, el módulo de ejecución de procedimiento configurado para:

- 55 recibir solicitudes de ejecución desde el coordinador de base de datos paralela;
 ejecutar procedimientos almacenados correspondientes, incluyendo los procedimientos almacenados correspondientes una o más sentencias de SQL y al menos otra sentencia no de SQL, que incluye:

enviar la una o más sentencias de SQL al coordinador de base de datos paralela para ejecución; y ejecutar la al menos otra sentencia no de SQL en el servidor de base de datos independiente.

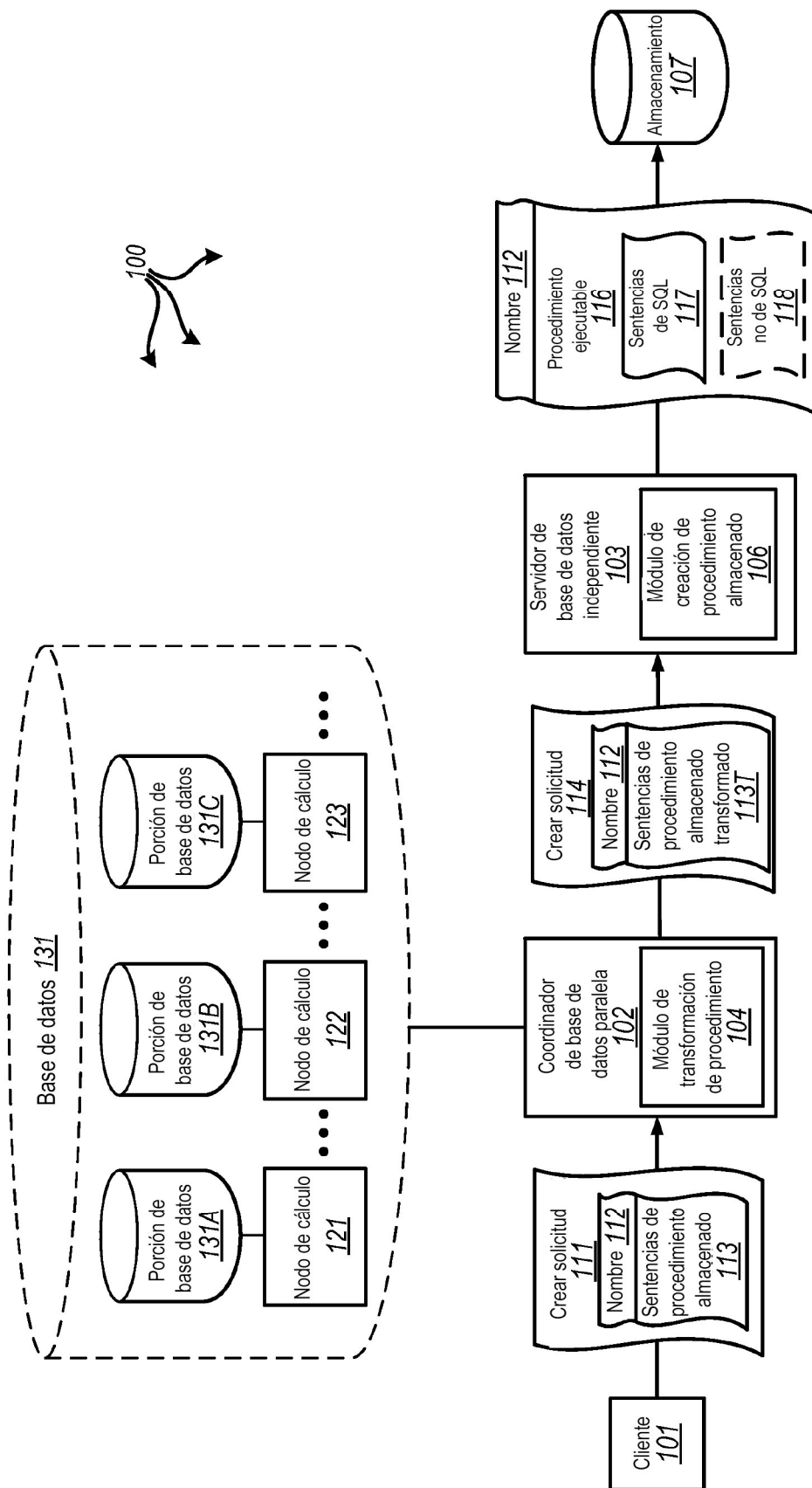


FIG. 1A

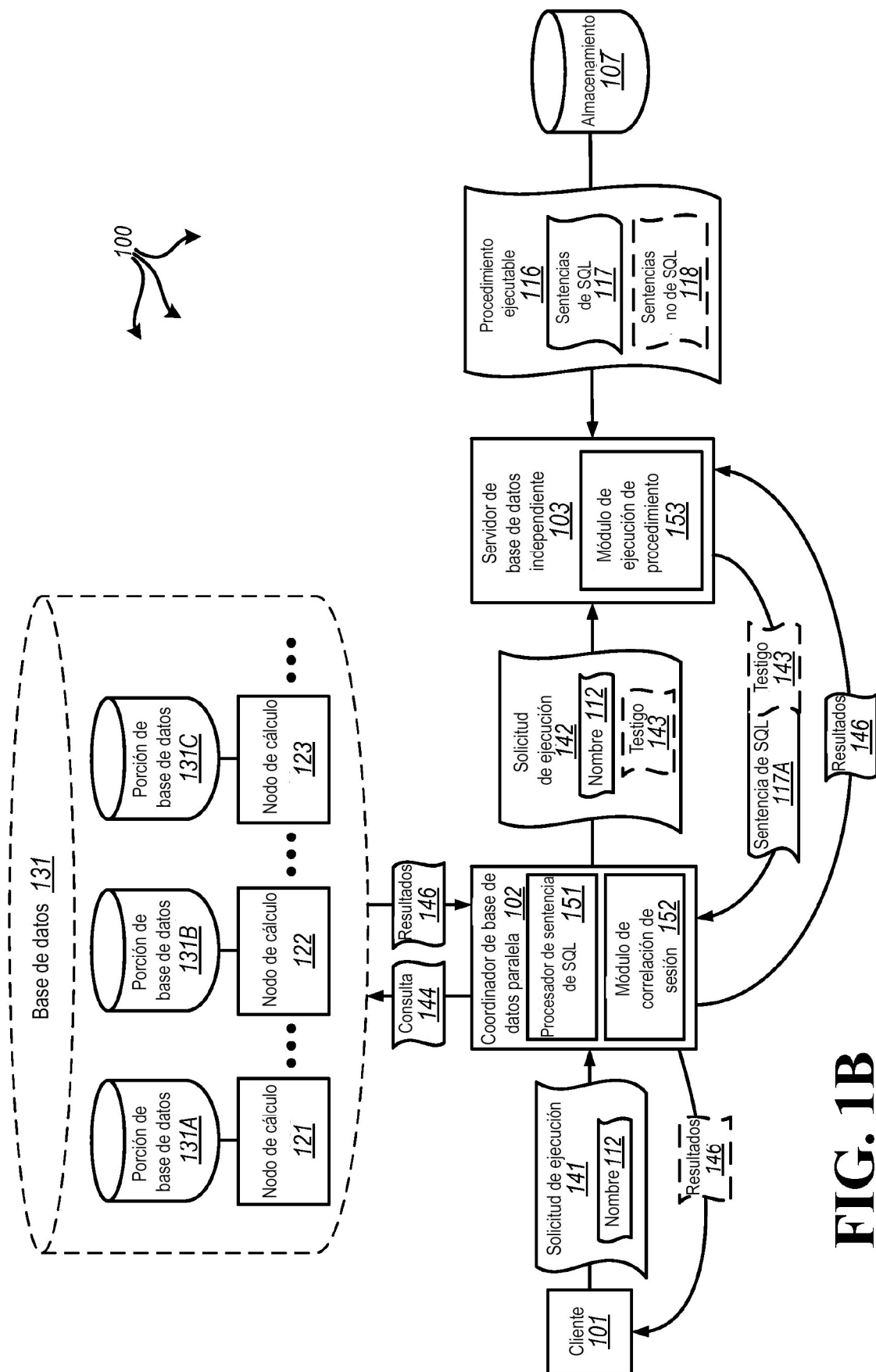


FIG. 1B

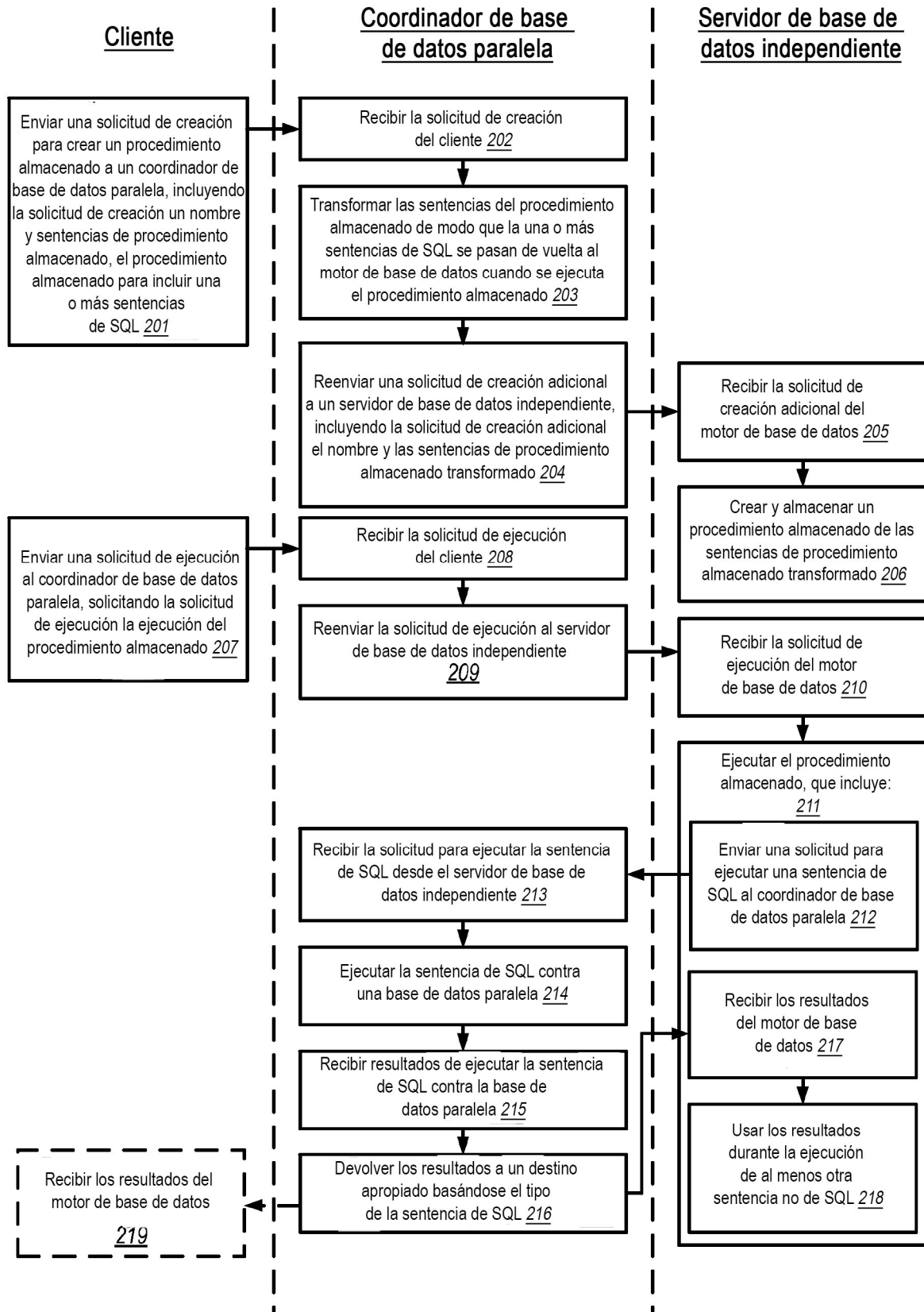


FIG. 2

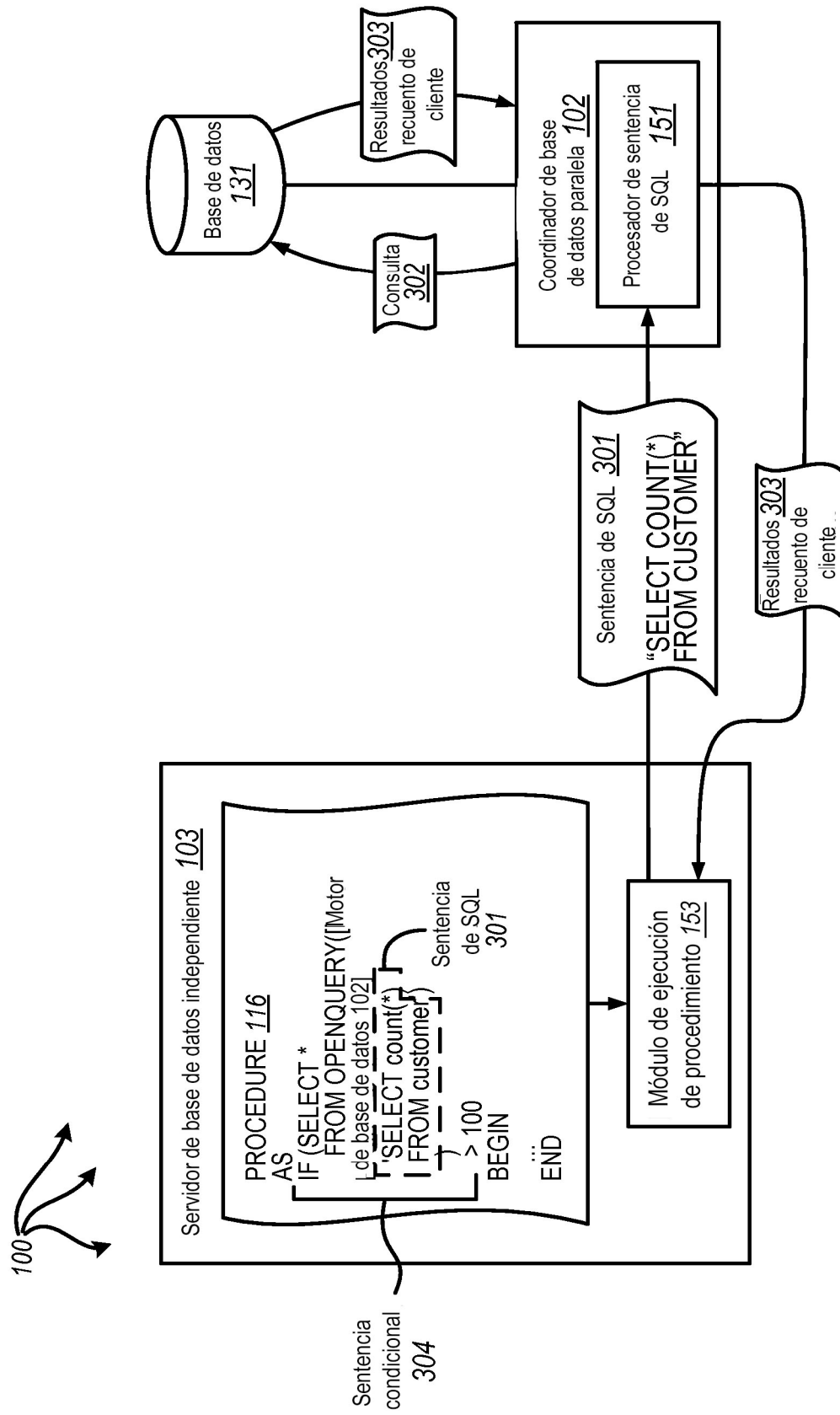


FIG. 3

Original	Después de transformación
<p>Consulta SELECT sin ningún parámetro</p> <pre>CREATE PROCEDURE test1 AS SELECT * FROM customer</pre>	<pre>CREATE PROCEDURE test1 AS EXECUTE ('SELECT * FROM customer') AT [LINKED DB]</pre> <p>Como alternativa, puede usarse OPENQUERY</p> <pre>CREATE PROCEDURE test1 AS SELECT * FROM OPENQUERY ([LINKED DB], 'SELECT * FROM customer') A</pre>

FIG. 4A

Original	Después de transformación
<p>Consulta SELECT con parámetros</p> <pre>CREATE PROCEDURE test1 AS SELECT * FROM customer WHERE c_custkey = @cid</pre>	<p>Puesto que esta consulta tiene un parámetro, se usa EXECUTE:</p> <pre>CREATE PROCEDURE test1 AS EXECUTE ('SELECT * FROM customer WHERE c_custkey = ?',@cid) AT [LINKED DB]</pre>

FIG. 4B

Original	Después de transformación
<p>Consulta SELECT con asignaciones variables pero no parámetros</p> <pre>CREATE PROCEDURE test1 AS SELECT @count = count(*) FROM customer</pre>	<pre>CREATE PROCEDURE test1 AS EXECUTE ('SELECT ? = count(*) FROM customer ', @count OUTPUT) AT [LINKED DB]</pre> <p>Como alternativa, puede usarse OPENQUERY</p> <pre>CREATE PROCEDURE test1 AS SELECT @count = alias1 FROM OPENQUERY ([LINKED DB], 'SELECT count(*) alias1 FROM customer')</pre>

FIG. 4C

Original	Después de transformación
<p>Subconsulta escalar sin parámetros</p> <pre>CREATE PROCEDURE test1 AS SET @count = SELECT count(*) FROM customer</pre>	<pre>CREATE PROCEDURE test1 AS SET @count = SELECT * FROM OPENQUERY([LINKED DB], 'SELECT count(*) FROM customer') A</pre>

FIG. 4D

Original	Después de transformación
<pre>CREATE PROCEDURE test1 AS IF (SELECT count(*) FROM customer) > 100 BEGIN ... END</pre>	<pre>CREATE PROCEDURE test1 AS IF (SELECT * FROM OPENQUERY([LINKED DB], 'SELECT count(*) FROM customer')) > 100 BEGIN ... END</pre>

FIG. 4E

Original	Después de transformación
<pre>CREATE PROCEDURE test1 AS SELECT * INTO temptable FROM customer</pre>	<pre>CREATE PROCEDURE test1 AS EXECUTE('SELECT * INTO temptable FROM customer') AT [LINKED DB]</pre>

FIG. 4F